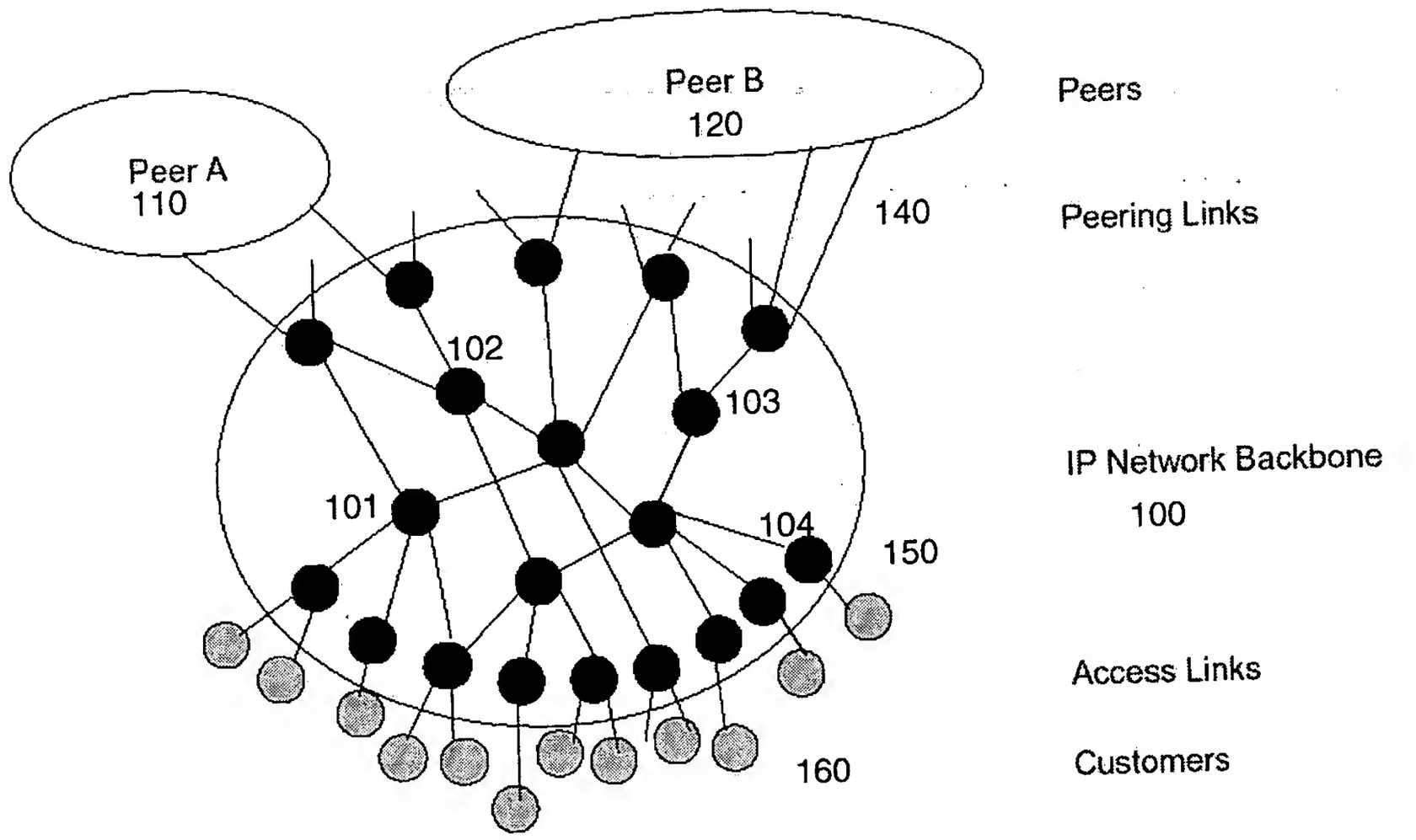
Peer B
120
Peer A
110
140
102
103
101
104
150
160
Peers
Peering Links
IP Network Backbone
100
Access Links
Customers

FIG. 1

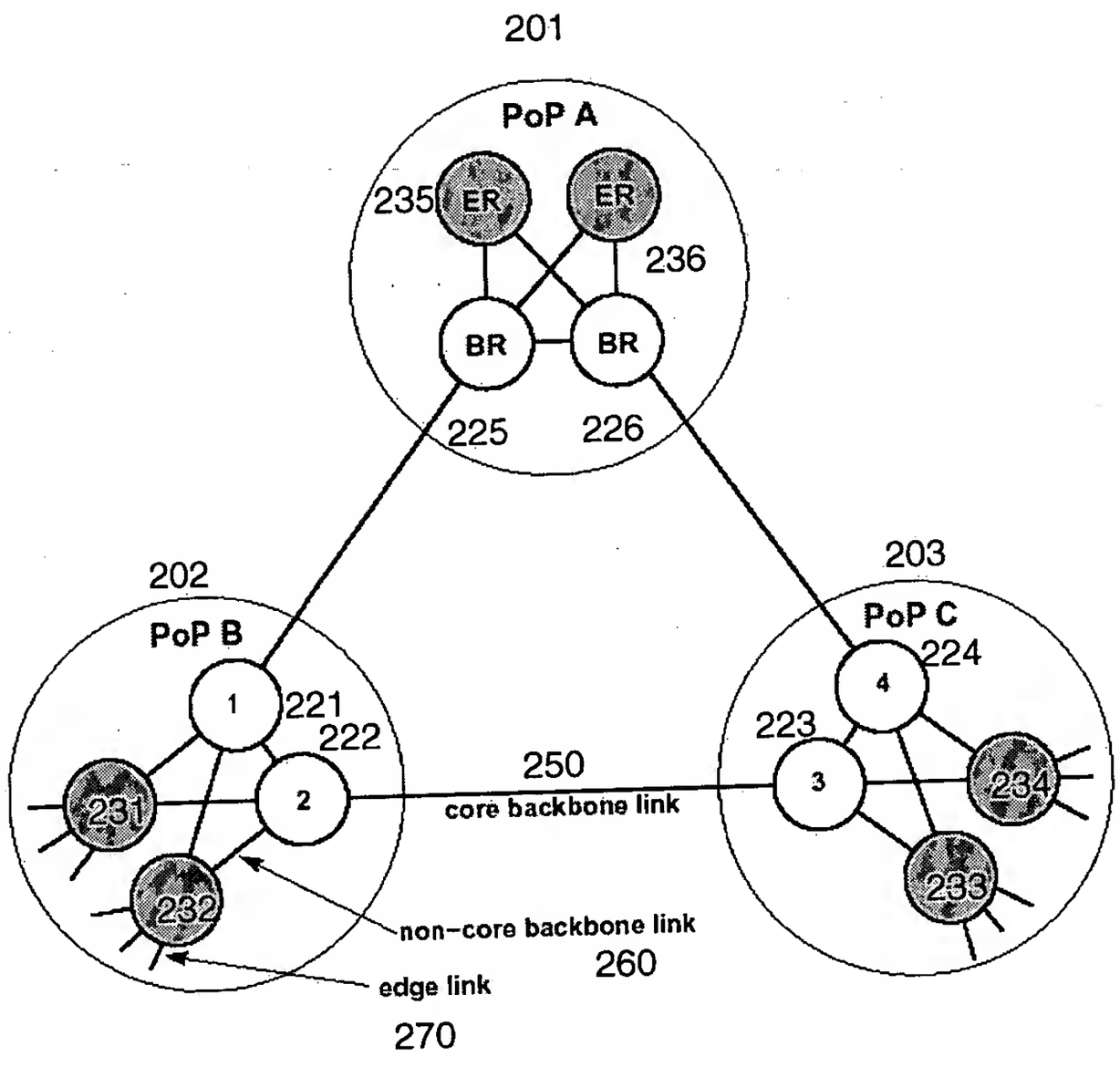
201
PoP A
235
ER
ER
236
BR
BR
225
226
202
PoP B
1
221
222
231
2
232
250
core backbone link
203
PoP C
224
4
223
3
234
233
non-core backbone link
260
edge link
270

FIG. 2

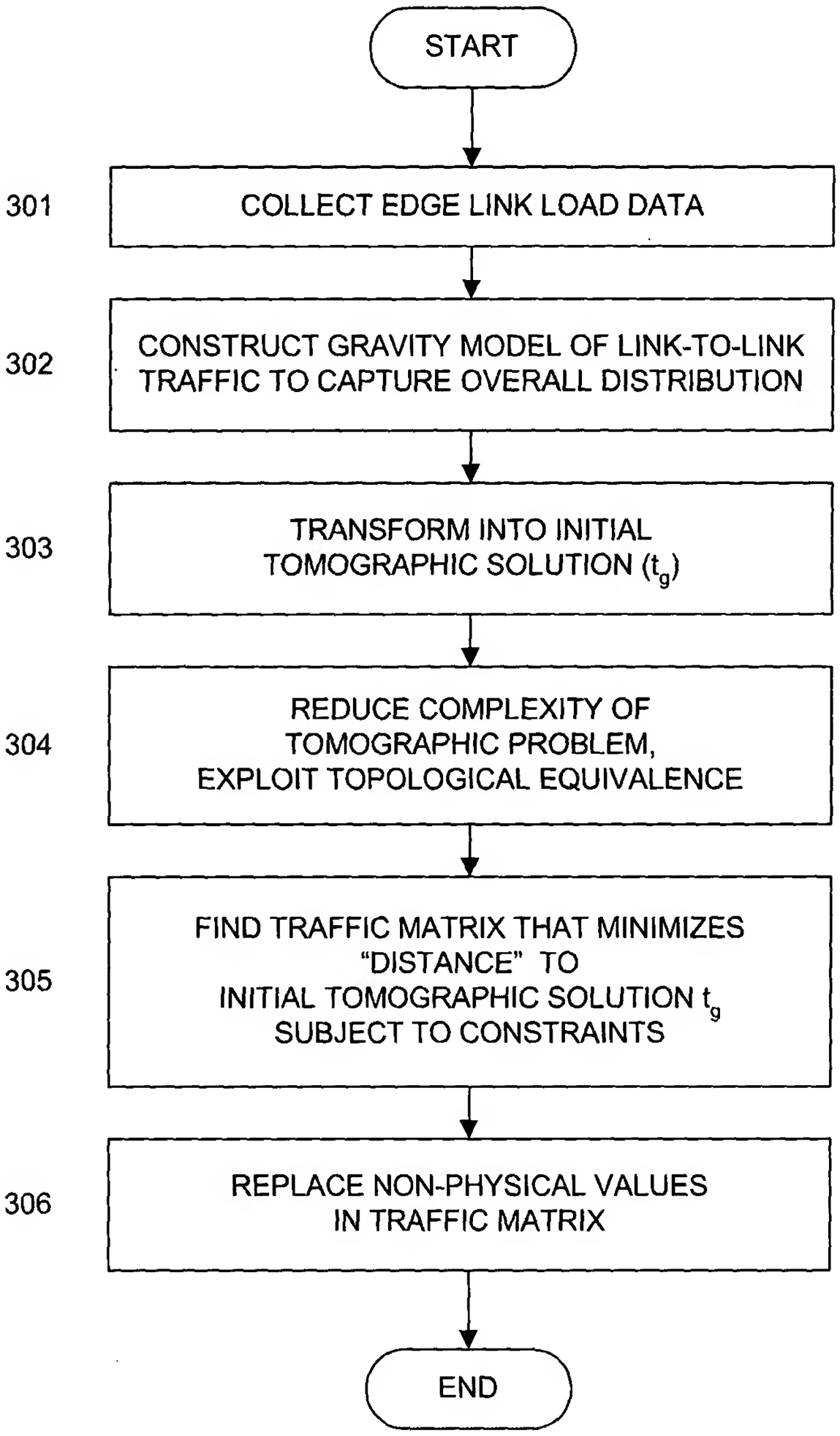
START
301
COLLECT EDGE LINK LOAD DATA
302
CONSTRUCT GRAVITY MODEL OF LINK-TO-LINK TRAFFIC TO CAPTURE OVERALL DISTRIBUTION
303
TRANSFORM INTO INITIAL TOMOGRAPHIC SOLUTION (t_g)
304
REDUCE COMPLEXITY OF TOMOGRAPHIC PROBLEM, EXPLOIT TOPOLOGICAL EQUIVALENCE
305
FIND TRAFFIC MATRIX THAT MINIMIZES "DISTANCE" TO INITIAL TOMOGRAPHIC SOLUTION t_g SUBJECT TO CONSTRAINTS
306
REPLACE NON-PHYSICAL VALUES IN TRAFFIC MATRIX
END

FIG. 3

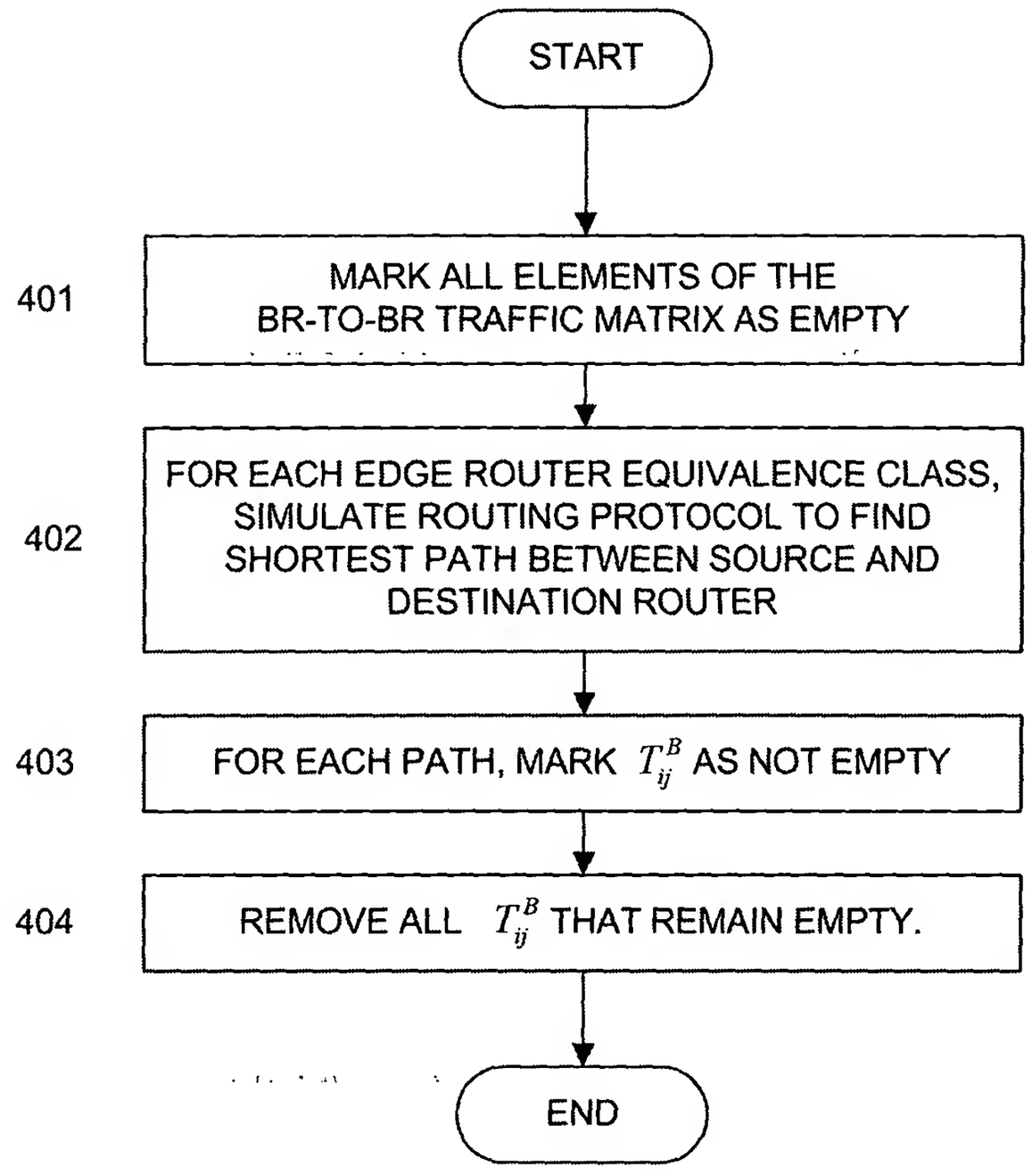
START
401
MARK ALL ELEMENTS OF THE BR-TO-BR TRAFFIC MATRIX AS EMPTY
402
FOR EACH EDGE ROUTER EQUIVALENCE CLASS, SIMULATE ROUTING PROTOCOL TO FIND SHORTEST PATH BETWEEN SOURCE AND DESTINATION ROUTER
403
FOR EACH PATH, MARK T_{ij}^{B} AS NOT EMPTY
404
REMOVE ALL T_{ij}^{B} THAT REMAIN EMPTY.
END

FIG. 4

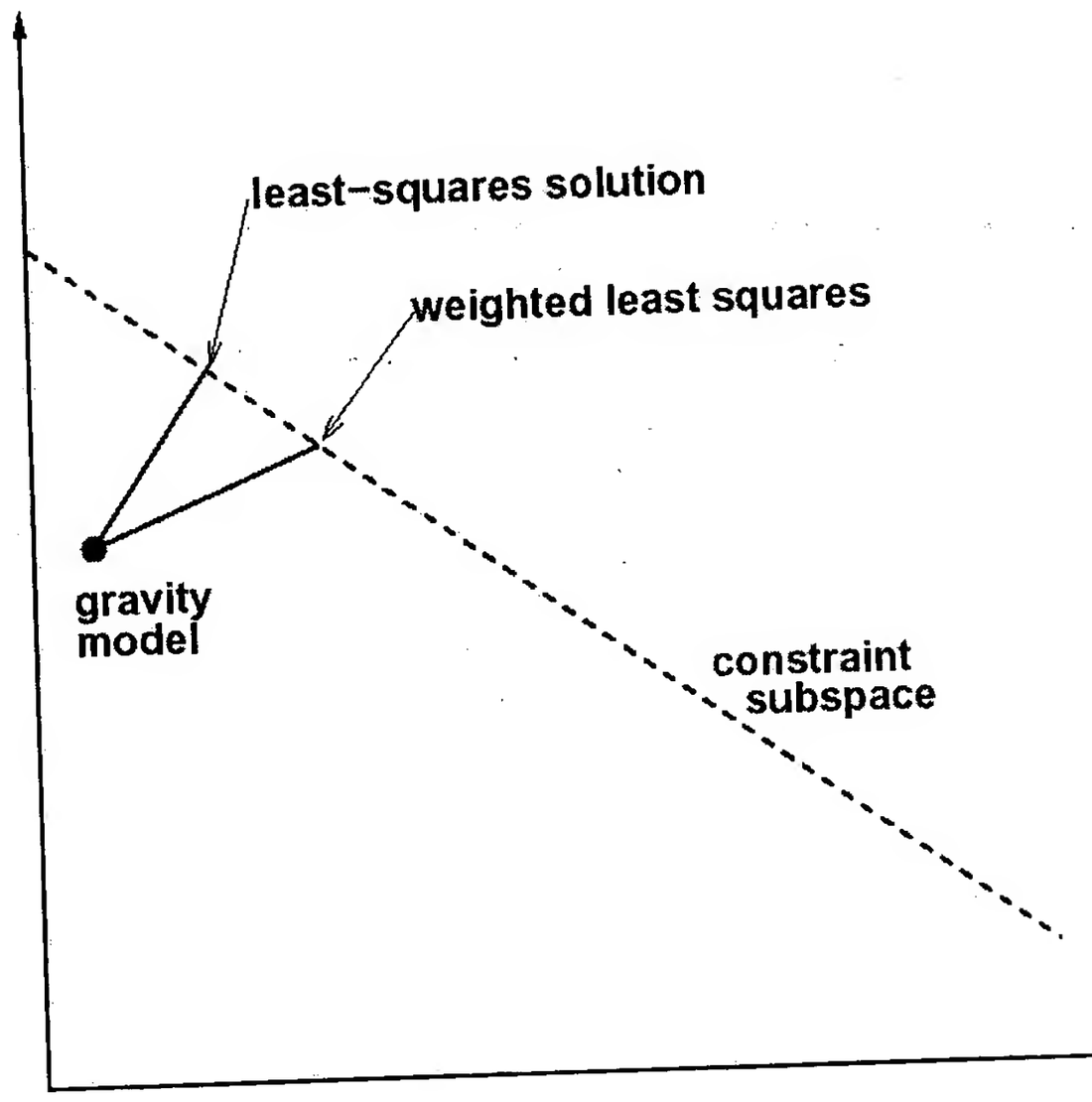
least-squares solution
weighted least squares
gravity
model
constraint
subspace

FIG. 5

```
%
%  weighted least-squares estimate of
%  the traffic matrix
%
%  Input:
%     A     matrix A in constraints x=A*t
%     x     vector x in constraints x=A*t
%     tg    initial gravity model solution
%     w     weight vector
%
%  Output:
%     t     estimated traffic matrix (as a
%           vector) that minimizes |(t-tg)./w|
%           among all t's that minimize |A*t-x|
%
function [t] wlse(A,x,tg,w)

   %  equivalently transform x=A*t into
   %  xw=Aw*tw, where tw=(t-tg)./w
   xw = x - A*tg;
   [r, c] = size(A);
   Aw = A .* repmat(w', r, 1);

   %  solve tw=Aw*tw by computing the pseudo-
   %  inverse of matrix Aw (through svd)
   tw = pinv(full(Aw)) * xw;

   %  transform tw back to t
   t = tg + w .* tw;
```

FIG. 6